

Analizați întreruperile cu serviciile lor:

Tabel. 1.2. Întreruperi folosite pentru afișarea datelor pe ecran

Tabel. 1.1. Întreruperi folosite pentru preluarea datelor de la tastatură	
Întreruperi folosite pentru preluarea datelor de la tastatură	Exemplu de apel:
Preluare șir de caractere	
Varianta I) cu int 21h, serviciul 0Ah; ; în DX să fie adresa de început a șirului unde s-a alocat spațiu în memorie pentru el De exemplu: sir db 20,22 dup (?)	mov dx, offset sir mov ah,0Ah int 21h
Preluare caracter în AL	
Varianta II) cu int 16h, serviciul 0h; preluare fără ecou	mov ah,0h int 16h
Varianta III) cu int 21h, serviciul 01h; preluare cu ecou	mov ah,01h int 21h
Varianta IV) cu int 21h, serviciul 08h; preluare fără ecou	mov ah,08h int 21h

Întreruperi folosite pentru afișarea datelor pe ecran		Exemplu de apel:
Afișare șir de caractere		
Varianta I) ; în DX e adresa de început a șirului cu int 21h, serviciul 09h; mesaj - terminat cu caracterul '\$'		mov dx, offset sir mov ah,09h int 21h
Afișare caracter din DL		
Varianta II) cu int 21h, serviciul 02h; se afișează caracterul din registrul DL		mov dl, 'A' mov ah,02h int 21h
Afișare caracter din AL		
Varianta III) cu int 10h, serviciul 0Eh; se afișează caracterul din registrul AL (mod teletype)		mov al, 'A' mov ah,0Eh int 10h
Varianta IV) cu int 10h, serviciul 09h și poziționare cursor cu int 10h, serviciul 02h.		mov dl,0 mov dh,0 mov ah,02h int 10h ; cursor
		mov al, 'A' mov ah,09h mov bl, 00100001b mov cx,1 int 10h

Problemă rezolvată:

1. Sa se scrie un program minimal care sa afiseze pe ecran (folosind int 21h cu serviciul 02h) un caracter 'A' pe ecran:

A1) puneti caracterul 'A' direct in registru;

A2) definiti caracterul 'A' in memorie intr-o variabila numita val;

A3) preluati caracterul de la tastatura (folosind int 16h cu serviciul 0).

A1) Versiunea cu definire de dată direct în registru	A2) Versiunea cu definire de dată în memorie	A3) versiune cu preluare de la tastatura
org 100h ; a1 .data .code ;depune în DL codul Ascii al caracterului ; ce se dorește afișat ('A') mov dl, 'A' ; DL='A' ; serviciul de afișare caracter din DL mov ah,02h ; serviciul 02h int 21h ; apel int 21h ret	org 100h ; a2 .data Val db 'A' ; definește o variabilă ; de tip byte în memorie .code ;preia Val din memorie în DL mov dl, Val ; DL='A' ; serviciul de afișare caracter din DL mov ah,02h ; serviciul 02h int 21h ; apel int 21h ret	org 100h ; a3 .data .code mov ah, 0 int 16h ; caracterul preluat de la ; tastatura se va regăsi în AL mov dl, al ;trebuie copiat în DL ; serviciul de afișare caracter din DL mov ah,02h ; afișare int 21h ret

Sugestie: Exersați prin înlocuirea în varianta A3) cu alte servicii de preluare de la tastatura și afișare pe ecran a unui singur caracter.

Probleme propuse :

(0.25p)1. Scrieti un caracter A pe ecran cu o culoare albastru pe fond galben, in mijlocul ecranului

(0.25p)2. Scrieti 3 caractere AAA pe ecran cu o culoare verde pe fond mov, in coltul dreapta jos al ecranului

3. Scrieti un mesaj 'Ana are mere' in segmentul de date si afisati-l pe ecran

(0.25p) a) folosind int 21h, serv 09h

(0.5p) b) folosind oricare alt serviciu

(0.5p) 4. Scrieti o secventa care sa preia de la tastatura un text (de exemplu se va tasta Ana are mere) si sa-l depuna in memorie intr-o variabila numita text.

(+0.25) afisati fiecare caracter al mesajului cu un alt atribut de culoare

(0.5p) 5. Sa se scrie o secventa de program care sa preia de la tastatura 2 numere, sa realizeze suma lor si sa afiseze aceasta suma pe ecran. (se vor da numere a.i. suma lor sa fie <9)

6. Explicați de ce urmatoarea problema afiseaza sirul in ordine inversa?

```
org 100h
.data
sir db 'Ana are mere'
I EQU $-sir
.code
mov si, 0
mov cx, I
mov BL, 0Fh
et:
    mov AL, sir[SI]
    add BL, 11h
    mov AH, 09h
    int 10h
    inc SI
loop et
```



7. Modificati problema de la punctul 6) astfel incat sirul sa fie afisat corect pe ecran



Problema 5:

Rezolvata fara subrutine:

```
org 100h ; 05
.data
a db ? ; se alocă doar spațiu în memorie pentru variabila a, dar nu se definește
b db ? ; se alocă doar spațiu în memorie pentru variabila b, dar nu se definește
suma db ? ; se alocă doar spațiu pentru variabila suma, dar nu se definește
.code
; preluarea primului număr de la tastatură
mov ah,0 ; serviciul de preluare tastă de la tastatură
int 16h ; întreruperea 16h cu serviciul 0h (în AH)
; dacă de exemplu se va apăsa tasta 2, în AL se va returna
; codul Ascii al lui 2 => AL=32h
sub al,30h ; pentru a obține valoarea 2, se va scădea 30h din AL

mov a,al ; se mută în memorie la locația rezervată pentru a

; preluarea celui de-al doilea număr de la tastatură
mov ah,0 ; serviciul de preluare tastă de la tastatură
int 16h ; întreruperea 16h cu serviciul 0h (în AH)
; dacă de exemplu vom apăsa tasta 6, în AL se va returna
; codul Ascii al lui 6 => AL=36h
sub al,30h ; pentru a obține valoarea 6, vom scădea 30h din AL

mov b,al ; se mută în memorie la locația rezervată pentru b

add al,a ; în AL este deja valoarea b, deci peste ea se adaugă a=> AL=8
mov suma,al ; se mută valoarea sumei în mem. la locația rezervată pt. suma

add al,30h ; se transformă valoarea de afișat în cod Ascii => AL=38h
mov ah,0Eh ; se afișează folosind serviciul 0Eh, mod teletype
int 10h ; apel întrerupere int 10h

ret ; revenire în S.O.
```

Problema rezolvata cu subrutina :

raportat la varianta anterioară, s-au luat instrucțiunile specificate îngroșat și s-au creat 2 subrutine (**preiaCifra**- cu cele 3 instrucțiuni de la începutul programului și **afisCifra** – cu cele 3 instrucțiuni de la sfârșit). Structura unei proceduri se specifică prin construcțiile PROC și ENDP, apelul prin instrucțiunea CALL, iar revenirea prin RET.

```
org 100h ; 05
.data
a db ? ; se alocă doar spațiu în memorie pentru variabila a, dar nu se definește
b db ? ; se alocă doar spațiu în memorie pentru variabila b, dar nu se definește
suma db ? ; se alocă doar spațiu pentru variabila suma, dar nu se definește
.code ; 05_2

call preiaCifra ; prin apelul acestei subrutine s-a preluat prima cifră
; de exemplu, în AL s-a obținut 2 dacă s-a apăsât tasta '2'
mov a,al ; se mută în memorie la locația rezervată pentru a

call preiaCifra ; prin apelul acestei subrutine s-a preluat cea de-a doua
cifra
; de exemplu, în AL s-a obținut 6 dacă s-a apăsât tasta '6'
mov b,al ; se mută în memorie la locația rezervată pentru b

add al,a ; AL=a+b
mov suma,al ; suma=a+b=2+6=8

call afisCifra ; se va afișa valoarea 8 pe ecran

ret ; revenire în S.O.
```

în simulator, abia după ref se vor așeza procedurile

; procedura de mai jos nu se va putea folosi și pentru litere, alte caractere, etc
; ea va funcționa doar pentru cifre întrucât se scade 30h – specific cifrelor
; în AL se va depune valoarea numărului preluat de la tastatură

```
preiaCifra proc ; se va considera că în AL returnează valoarea,
mov ah,0 ; serviciul de preluare tastă de la tastatură
int 16h ; întreruperea 16h cu serviciul 0h (în AH)
; dacă, de exemplu, se va apăsa tasta 2, în AL se va
; returna codul Ascii al lui 2=> AL=32h
sub al,30h ; pentru a obține valoarea 2, se va scădea 30h din AL
ret ; revenire din surutină
```

preiaCifra endp

; procedura de mai jos nu se va putea folosi și pentru litere, alte caractere, etc
; ea va funcționa doar pentru cifre întrucât se adună 30h – specific cifrelor.
; anterior apelului, în AL se va depune valoarea numărului ce se dorește a fi afișat

afisCifra proc ; se va considera că în AL primește valoarea de afișat

```
mov ah,0Eh ; serviciul de afișare mod teletype
add AL,30h ; se formează cod Ascii necesar afișării
int 10h ; întreruperea 10h cu serviciul 0Eh (în AH)
ret ; revenire din surutină
```

afisCifra endp

Problema propusa:

Adaptati problema astfel incat in locul subrutinelor sa folositi macouri .